7u

UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/797,886	03/09/2004	Bratin Saha	10559/913001/P18139/Intel	5086

20985	7590	09/21/2006

FISH & RICHARDSON, PC
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER
JOHNSON, BRIAN P.

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 09/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

		Application No.	Applicant(s)
Office Action Summary		10/797,886	SAHA, BRATIN
		Examiner	Art Unit
		Brian P. Johnson	2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>05 July 2006</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-34* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-34* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

        1.☐ Certified copies of the priority documents have been received.

        2.☐ Certified copies of the priority documents have been received in Application No. _____.

        3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

1.   Claims 1-34 have been examined.

Acknowledgment of papers filed: amendments and remarks on 05 July 2006.

The papers filed have been placed on record.

Specification

2.   The title is accepted. Objection is withdrawn.

Claim Objections

3.   Claim 26 is objected to because of the following informalities: in the final line,

perhaps the word "affect" is more appropriate than "effect". Please consider similar

claims in light of this objection. Appropriate correction is required.

Claim Rejections - 35 USC § 101

4.   Rejection is withdrawn.

Claim Rejections - 35 USC § 102

5.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6.   Claims 11, 12, 15, 16, 17, and 20 are rejected under 35 U.S.C. 102(b) as being

anticipated by Rajwar (Speculative Lock Elision).

7.   Regarding claim 11, Rajwar discloses a machine-implemented method

comprising: generating parallel processes in a data processing machine (Rajwar page

294 col 1 section 1); effecting synchronization between the parallel processes using

processor speculation in the data processing machine (Rajwar page 295 col 1) to

speculatively read-modify-write a lock variable associated with a critical section (Rajwar

page 295 col 1); and providing output resulting from the synchronized parallel

processes (Rajwar page 294 col 2 second paragraph).

Note the use of a store instruction. This is considered to be a readable output.

8.   Regarding claim 12, Rajwar discloses the method of claim 11, wherein said

generating parallel processes comprises running a software program (Rajwar page 294

figure 1) that spawns multiple threads in the data processing machine (Rajwar page 294

section 1 col 1).

9.   Regarding claim 15, Rajwar discloses the method of claim 11, wherein said

providing output comprises sending the output to another data processing machine

(Rajwar page 294 section 1 col 1).

Note that the store instructions within a processor stores data to memory. This output is then picked up by another processor in the "shared memory multiprocessors" disclosed in Rajwar.

10.   Regarding claim 16, Rajwar discloses an article comprising a machine-readable storage medium embodying information indicative of instructions that when performed by one or more machines result in operations (Rajwar page 294 col 2 fig 1)

Note that clearly, the instructions disclosed in figure 1 must be contained on a computer readable medium for the processing system to execute the instructions.

Comprising: generating parallel processes in a data processing machine (Rajwar page 294 col 1 section 1); effecting synchronization between the parallel processes using processor speculation in the data processing machine to speculatively read-modify-write a lock variable associated with a critical section (Rajwar page 295 col 1); and providing output resulting from the synchronized parallel processes (Rajwar page 294 col 2 "store instruction").

11.   Regarding claim 17, Rajwar discloses the article of claim 16, wherein said generating parallel processes comprises running a software program that spawns multiple threads in the data processing machine (Rajwar page 294 section 1 col 1).

12.   Regarding claim 20, Rajwar discloses the article of claim 16, wherein said

providing output comprises sending the output to another data processing machine

(Rajwar page 294 section 1 col 1).

     Note: see claim 15

13.   Claims 11, 15, 16, and 20 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Transactional Memory (herein Moss).

14.   Regarding claim 11, Moss discloses a machine-implemented method comprising:

generating parallel processes in a data processing machine (Moss page 290 sections

2.1 and 2.2); effecting synchronization between the parallel processes (page 289 col 2

"lock-free synchronization") using processor speculation in the data processing machine

(Moss page 290 section 2.2) to speculatively read-modify-write a lock variable

associated with a critical section (Moss page 289 col 2); and providing output resulting

from the synchronized parallel processes (Moss page 290 section 2.1).

     Note the use of a store instruction. This is considered to be a readable output.

15.   Regarding claim 15, Rajwar discloses the method of claim 11, wherein said

providing output comprises sending the output to another data processing machine

(Rajwar page 294 section 1 col 1).

Note that the store instructions within a processor stores data to memory. This output is then picked up by another processor in the "shared memory multiprocessors" disclosed in Rajwar.

16. Regarding claim 16, Moss discloses an article comprising a machine-readable storage medium embodying information indicative of instructions that when performed by one or more machines result in operations (Moss page 290 section 2.1)

Note that clearly, the instructions disclosed in section 2.1 must be contained on a computer readable medium for the processing system to execute the instructions.

Comprising: generating parallel processes in a data processing machine (Rajwar page 294 col 1 section 1); effecting synchronization between the parallel processes using processor speculation in the data processing machine to speculatively read-modify-write a lock variable associated with a critical section (Moss page 290 section 2.2); and providing output resulting from the synchronized parallel processes (Moss page 290 section 2.1).

17. Regarding claim 20, Moss discloses the article of claim 16, wherein said providing output comprises sending the output to another data processing machine (Moss page 290 section 2.1).

Note: see claim 15

Claim Rejections - 35 USC § 103

18.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

19.   Claims 1, 6, 7, 14, and 19 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Transactional Memory (herein Moss) in view of Lam (Enhancing

Software Reliability with Speculative Threads).

20.   Regarding claim 1, Moss discloses a processor (page 289 abstract) comprising:

a front end that obtains instructions (see below);

Note that the terms, "front end" and "back end", as vaguely defined are limited

only by how the reference is defined. Any portion of the processor can be considered a

"front end" or "back end".

And a back end that provides speculative execution of the instructions (page 290

section 2.2);

Moss also discloses synchronization of parallel processes (page 289 column 2—

"lock-free synchronization")

Moss fails to disclose an instruction that follows all claimed limitations.

Lam discloses an instruction set architecture including speculative execution

control circuitry that handles at least one machine instruction that facilitates

synchronization between parallel processes by exposing the processor speculation to

program control (Lam page 187 TRY instruction).

Although it is clear from Moss that synchronization is maintained, it does not

describe how the start of the critical section is determined. Lam, however, uses this

TRY instruction which "allows the program to recover from attacks that overwrite data

structures beyond those expected". Moss would be motivated to use this technique

because it gives a large amount of control to the programmer, who can remove

unnecessary processor commands (which happened to be the general motivation

behind the Moss invention initially). Additionally, for the same motivation, Moss would

be motivated to include other elements of lock speculation in an explicit instruction.

It would have been obvious at the time of the invention for one of ordinary skill in

the art to allow the processing system of Moss maintain synchronization through explicit

instructions such as TRY.

21.   Regarding claim 6, Moss/Lam discloses the processor of claim 1, wherein the at

least one machine instruction comprises: a speculative execution instruction that takes

first and second operands,

Note that the term "operands" most commonly refers to inputs of an arithmetic

operation. Applicant's invention does not use the claimed operands for that purpose.

Consequently, Examiner believes it is reasonable to consider the claimed operands to

be any register associated to the instruction.

Behaves as a no-op if a memory location indicated by the first operand contains

a first value (Lam page 187 section 3.2),

Note that the TRY instruction serves no purpose if there is no error.

Causes the processor to speculatively execute additional instructions if the

memory location contains a second value (Moss page 290 sections 2.1 and 2.2), and

causes the processor to start executing instructions from an address indicated by the

second operand if a mis-speculation occurs (Lam page 187 TRY instruction);

Note the {addr} value.

And a speculation termination instruction that takes first and second operands

and causes the processor to begin retiring the additional instructions if the additional

instructions have been speculatively executed (Moss page 290 COMMIT instruction).

22.   Regarding claim 7, Moss/Lam discloses the processor of claim 6, wherein the

mis-speculation comprises an interrupt (Lam page 187 section 3.2 first column).

Note "error-detected", which will interrupt the speculative execution.

23.   Regarding claim 14, Moss/Lam discloses the method of claim 11, wherein said

effecting synchronization comprises placing in an out-of-order execution management

unit of a processor (see below),

Note that the use of speculative lock acquisition is used to facilitate out-of-order

execution

At least one machine instruction that limits when other machine instructions are

retired from the out-of-order execution management unit (Moss page 290 COMMIT and

ABORT instructions).

24.   Regarding claim 19, Moss/Lam discloses the article of claim 16, wherein said

effecting synchronization comprises placing in an out-of-order execution management

unit of a processor (see below),

Note that the use of speculative lock acquisition is used to facilitate out-of-order

execution

At least one machine instruction that limits when other machine instructions are

retired from the out-of-order execution management unit (Moss page 290 COMMIT and

ABORT instructions).

25.   Claims 21-25, 27, and 31-34 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Transactional Memory (herein Moss) in view of Lam (Enhancing

Software Reliability with Speculative Threads) in view of Rajwar.

26.   Regarding claim 21, Moss/Lam discloses a machine-implemented method

comprising: speculatively executing machine instructions (Moss page 290 section 2.1),

including a memory access instruction (Moss page 290 section 2.1),

Note the use of load and test instructions.

In a processing system to effect synchronization between parallel processes

(Moss page 289 col 2); retiring the speculatively executed machine instructions (Moss

page 290 COMMIT instruction); and maintaining cache coherence in the processing

system during said executing and said retiring to identify a mis-speculation to effect the

synchronization between the parallel processes (Moss page 290 COMMIT instruction),

wherein the speculatively executing comprises performing a speculative read-modify-write associated with a critical section (Moss page 289 col 2), retiring the speculatively executed machine instructions (Moss page 290 COMMIT instruction), maintaining cache coherence in the processing system during said executing and said retiring to identify a mis-speculation to effect the synchronization between the parallel processes (Moss page 290 section 2.1).

Moss/Lam fails to disclose that the speculation is to a lock variable.

Rajwar discloses a lock variable speculatively executed (without the lock) and re-executed (with the lock) during a failure (page 295 col 1).

Examine asserts that speculative lock execution can cause deadlocks that cause a program to constantly re-execute information based on critical section contention. Moss/Lam discloses re-executing instructions during such a contention, but doesn't how this re-execution occurs. Moss/Lam would have been motivated to utilize the lock variable, as shown in Rajwar, to prevent a mistake from repeating itself while still allowing the speculative nature of the critical section.

It would have been obvious at the time of the invention for one of ordinary skill in the art to take the system of Moss/Lam and include a lock variable (as shown in Rajwar) that is attained after a mis-prediction of the speculative execution shown in both Rajwar and Moss/Lam.

27.   Regarding claim 22, Moss/Lam/Rajwar discloses the method of claim 21,

wherein said maintaining cache coherence comprises providing invalidation based

cache coherence (Moss page 290 section 2.1)

28.   Regarding claim 23, Moss/Lam/Rajwar discloses the method of claim 21,

wherein said speculatively executing machine instructions comprises speculatively

executing machine instructions in the processing system comprising multiple

processors, and the mis-speculation comprises a memory dependency violation (Moss

page 289 Abstract).

29.   Regarding claim 24, Moss/Lam/Rajwar discloses the method of claim 21,

wherein the mis-speculation comprises at least one of an interrupt (Lam page 187

section 3.2 first column.),

   Note "error-detected", which will interrupt the speculative execution.

   An external event (Lam page 187 section 3.2 first column.),

   Note that the "error-detected" is considered to be external to the speculation

logic.

   And a memory dependency violation (Moss page 290 section 2.1).

30.   Regarding claim 25, Moss/Lam/Rajwar discloses a system comprising: a

processor (Moss page 289 Abstract) having a processor architecture that provides

speculative execution of machine instructions (Moss page 290 section 2.2) and exposes

said speculative execution to program control through at least one machine instruction

(Lam page 187 TRY instruction); and a memory coupled with the processor, the

memory embodying information indicative of instructions (Moss page 290 section 2.1),

> *Note that for these instructions to be read, a memory must be used.*

Including the at least one machine instruction, that result in synchronization

between parallel processes when performed by the processor with detection of mis-

speculation (Lam page 197 TRY instruction), wherein performance of the instruction by

the processor comprises performing a speculative read-modify-write to the lock variable

associated with a critical section (MOSS page 289 col 2—in light of Rajwar).

31.   Regarding claim 27, Moss/Lam/Rajwar discloses the system of claim 25, wherein

the processor comprises a multiprocessor (Moss page 289 Abstract).

32.   Regarding claim 31, Moss/Lam/Rajwar discloses the system of claim 25, wherein

the at least one machine instruction comprises: a speculative execution instruction (Lam

page 187 TRY instruction) that takes first and second operands,

Behaves as a no-op if a memory location indicated by the first operand contains

a first value (Lam page 187 section 3.2), causes the processor to speculatively execute

additional instructions if the memory location contains a second value (Moss 290

sections 2.1 and 2.2), and causes the processor to start executing instructions from an

address indicated by the second operand if a mis-speculation occurs (Lam page 187

TRY instruction); and a speculation termination instruction that causes the processor to

begin retiring the additional instructions if the additional instructions have been

speculatively executed (Moss page 290 COMMIT instruction).

33.   Regarding claim 32, Moss/Lam/Rajwar discloses the system of claim 25, further

comprising an environmental sensor coupled with the processor (Lam page 187 end of

section 3.2).

    Note that the error detection if statement is considered to be an environmental

sensor.

34.   Regarding claim 33, Moss/Lam/Rajwar discloses a processing system

comprising: processing means for speculatively executing machine instructions in

response to a speculative execution instruction (Lam page 187 TRY instruction),

including means for detecting a mis-speculation (Moss page 290 COMMIT instruction);

means for treating multiple speculative instructions as a group for purposes of

retirement such that the multiple speculative instructions are flushed from the

processing means together (Moss page 290 COMMIT instruction) and execution

proceeds from an address in response to a detected mis-speculation to effect

synchronization between parallel processes (Lam page 187 TRY instruction and Moss

page 290 COMMIT instruction), wherein performance of the instructions by the

processing means comprises performing a speculative read-modify-write to a lock

variable associated with a critical section (Moss page 289 col 2—in light of Rajwar).

35.   Regarding claim 34, Moss/Lam/Rajwar discloses the processing system of claim

33, wherein said means for detecting a mis-speculation comprises means for

maintaining cache coherence in the processing means (Moss page 290 sections 2.1

and 2.2).

36.   Claims 2-5, and 8-10 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Moss/Lam in view of Christie (U.S. Patent No. 6,009,512).

37.   Regarding claim 2, Moss/Lam discloses the processor of claim 1.

       Moss/Lam contains an out-of-order processor (as suggested by the speculative

lock interface), but fails to disclose particular information about how the out-of-order

system works.

       Christie discloses a front end that comprises an in-order front end (Christie fig 5),

and the back end comprises an out-of-order execution engine (Christie fig 5), which re-

orders the instructions (Christie col 6 line 49), and one or more execution units that

perform the re-ordered instructions (Christie fig 5 references 1212A-C).

       For subsequent claims, Christie also discloses a branch prediction unit.

       Examiner asserts that out-of-order execution must start and end in program

order, while allowing the execution units execute out-of-order execution. Consequently,

an in-order front end with an out-of-order back end, as claimed, is likely an inherent

aspect of the invention for Moss/Lam; however, in case Examiner is neglecting to

consider an exception, this will be an obvious rejection. Moss/Lam would be motivated

to utilize an in-order front end with an out-of-order back end, because this is, as one of

ordinary skill in the art would have realized, a simple and efficient mechanism for out-of-

order processing.

Additionally, Moss/Lam would be motivated to utilize a branch prediction unit. As

one of ordinary skill in the art would have realized, a branch prediction unit can cause

large performance increases in a processor. An invention such as Moss/Lam with that

particular intention in mind would be particularly motivated to utilize this mechanism.

It would have been obvious at the time of the invention for one of ordinary skill in

the art for the processing system of Moss/Lam to utilize mechanisms of Christie that

were not described in great detail (so not to obscure the invention of Moss/Lam) such

as a branch prediction unit and Christie's out-of-order processing setup.

38.   Regarding claim 3, Moss/Lam/Christie discloses the processor of claim 2,

wherein the out-of-order execution engine comprises an out-of-order execution

management unit (Christie fig 5), including at least one buffer (Christie col 7 line 51),

and an in-order retire-store unit (Christie col 7 lines 54-64).

39.   Regarding claim 4, Moss/Lam/Christie discloses the processor of claim 3,

wherein the at least one buffer comprises a reorder buffer (Christie col 7 lines 54-64).

40.   Regarding claim 5, Moss/Lam/Christie discloses the processor of claim 1,

wherein the front end comprises a fetch-decode unit (see below)

Note that it is an inherent aspect of the referenced invention to have a fetch-decode unit. In order for the instructions to be processed, they must be fetched and decoded.

And a branch prediction unit (Christie col 6 line 35).

41.   Regarding claim 8, Moss/Lam/Christie discloses a processor comprising: a front end that obtains instructions; and a back end that provides speculative execution of the instructions (Christie fig 5) wherein the processor has an instruction set architecture including speculative execution control circuitry that handles a speculative execution instruction (Moss page 290 sections 2.1 and 2.2) and a speculation termination instruction (Lam COMMIT instruction).

42.   Regarding claim 9, Moss/Lam/Christie discloses the processor of claim 8, wherein the front end comprises an in-order front end, and the back end comprises an out-of-order execution engine (Christie Fig 5), which re-orders the instructions (Christie col 6 line 49), and one or more execution units that perform the re-ordered instructions (Christie fig 5 references 1212A-C).

43.   Regarding claim 10, Moss/Lam/Christie discloses the processor of claim 8, wherein the speculative execution instruction comprises an instruction that takes first and second operands,

Note: see claim 6.

Causes the processor to speculatively execute additional instructions if a

memory location contains a value (Lam page 187 section 1 col 1), and causes the

processor to start executing instructions from an address indicated by the second

operand if a mis-speculation occurs (Lam page 187 col 2 TRY {addr}), and the

speculation termination instruction comprises an instruction that causes the processor

to begin retiring the additional instructions (Moss page 290 COMMIT instruction).

44.   Claims 13 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Rajwar in view of common prior art.

45.   Regarding claim 13, Rajwar discloses the method of claim 11.

Rajwar fails to disclose synchronization comprises translating at least one high-

level software instruction into at least one machine instruction that controls speculative

execution in a processor.

Examiner asserts that the use of a high-level language is extremely common in

the art. Rajwar would be motivated to utilize this strategy to simply the code writing

process for a programmer at a high level language, and allow a compiler/assembler put

the language in machine code for the processor to read.

It would have been obvious at the time of the invention for one of ordinary skill in

the art to take the machine instructions in the processing system of Rajwar, and have

them translated from a high-level language.

46.   Regarding claim 18, Rajwar discloses the article of claim 16 as well as the

remaining limitations. See claim 13 above.

47.   Claim 28 is rejected under 35 U.S.C. 103(a) as being unpatentable over

Moss/Lam/Rajwar in view of common prior art.

48.   Regarding claim 28, Moss/Lam/Rajwar discloses the system of claim 27.

Moss/Lam fails to disclose that the multiprocessors are in a single die.

Examiner asserts that placing multiple processing units on a single die is

extremely common in the art and allows for faster, cheaper and more power effective

processing systems.

It would have been obvious at the time of the invention for one of ordinary skill in

the art to allow all the processors of Moss/Lam to be on a single die.

49.   Claim 26 is rejected under 35 U.S.C. 103(a) as being unpatentable over

Moss/Lam/Rajwar in view of Rajwar.

50.   Regarding claim 26, Moss/Lam discloses the system of claim 25.

Moss fails to disclose the use of a uniprocessor.

Rajwar discloses the use of a multithreaded uniprocessor (page 294 section 1 col

1)

Examiner asserts that Moss/Lam/Rajwar (as initially combined) would be motivated to utilize a multithreaded uni-processor system rather than a multiprocessor system. Examiner asserts that multithreaded systems have been shown to have substantial performance increases without the requirement of multiple processors— saving on area, power, and cost.

It would have been obvious at the time of the invention for one of ordinary skill in the art to implement the system of Moss/Lam as a uniprocessor (as shown in Rajwar)

51.   Claims 29 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Moss/Lam/Rajwar in view of common prior art.

52.   Regarding claims 29 and 30, Moss/Lam/Rajwar discloses the system of claim 25, further comprising: a communication interface (Rajwar page 294 section 1 col 1);

Note that the processing system is considered to be a communication interface.

Moss/Lam/Rajwar (as initially combined) fails to disclose a virtual machine, including a java virtual machine.

Examiner asserts that the use of a java virtual machine in a processing system is extremely common in the art and gives the processing system added security by preventing information inside the machine to be accessed outside the machine. Portability of code is another potential motivation.

It would have been obvious at the time of the invention for one of ordinary skill in the art to allow the system of Moss/Lam/Rajwar utilize a java virtual machine.

Response to Arguments

53.   Applicant's arguments filed 05 July 2006 have been fully considered but they are

not persuasive. Other arguments are considered moot under new grounds of rejection.

54.   Applicant state:

*"Independent claims 11 and 16 each recite, 'effecting synchronization between the parallel
processes using processor speculation in the data processing machine.' Thus, processor speculation is
used to bring about synchronization between the parallel processes. IN stark contrast, Rajwar describes
systems and techniques for avoiding synchronization altogether."*

Examiner disagrees. The use of lock speculation is not avoiding synchronization

for a variety of reasons. 1) Locks are not required for synchronization. This is the entire

point of both Applicant's invention and the referenced used for anticipation. Note the

use of the term "lock-free synchronization" in Moss. 2) Rajwar does obtain a lock when

the speculation is unsuccessful. 3) Applicant "avoids synchronization" in the same

sense that Applicant accuses Rajwar of doing the same. If avoiding synchronization

should truly be defined as Applicant contends, then Applicant's claims are not

appropriately claiming the invention disclosed in Applicant's specification.

55.   Applicant states:

*"Claims 11 and 16 have been amended to clarify the claimed subject matter and now recite
'...speculatively read-modify-write a lock variable associated with a critical section"...This amendment
makes clear that the speculative execution is of the lock acquisition code, not merely the critical section
as done in Rajwar."*

Examiner disagrees. The difference is not made clear by this amendment.

Rajwar speculatively executes a critical section (or, in other words, speculatively avoids

a read-modify-write lock) and, if mis-predicted, re-executes with the lock. This appears

to be precisely the same technique is disclosed in Applicant's Fig 5.

56.   Applicant states:

"Lam describes techniques to enhance the reliability of software through the use of monitoring programs executed speculatively in parallel with the main program being monitored...By definition, a monitoring program does not need to synchronize with the main program because the two programs doe not need to coordinate their accesses to a shared resource."

Examiner disagrees. Lam does discuss monitoring functions; however, the "TRY

instruction" and exception catching functionality is clearly used to affect synchronization

in the ways described in the combination.

57.   Applicant states:

"Rajwar actually teaches away from the proposed combination"

Examiner agrees. In light of this argument, a new grounds rejection has been
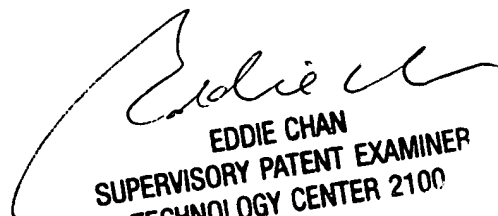
made.

Conclusion

58.   The following is text cited from 37 CFR 1.11(c): In amending in reply to a

rejection of claims in an application or patent under reexamination, the applicant or

patent owner must clearly point out the patentable novelty which he or she thinks the

claims present in view of the state of the art disclosed by the references cited or the

objections made. The applicant or patent owner must also show how the amendments

avoid such references or objections.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Brian P. Johnson whose telephone number is (571) 272-2678. The examiner can normally be reached on 8-4:30 M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100